

*Courtesy of:
Scitor Corporation
256 Gibraltar Drive • Sunnyvale, CA • 94089
800/533-9876*

Automatic Project Management: A Classic Oxymoron

We live in the age of automation. The coffee goes on by itself, each morning. The bread maker does it all: mixing, kneading, rising, baking ... just add the ingredients and press start. I have trouble finding a luxury automobile that does not have an automatic transmission (which I will not drive). Robots and N/C machines make most of our products. And, yes, **there are now some people who think that we want the project management process to be "automatic"**.

Not that there is a problem in seeking automation. But not the entire process. We have to draw the line somewhere. To qualify this, let's examine the components of a project management support system, and where automation fits in. First of all, there are two basic stages: when we plan the project, and when we progress it.

The Planning Stage

The basic steps here are to identify the overall project goals, milestones and strategy; to identify the work; to schedule the work; to assign resources to the work; to reschedule considering resources; and (optionally) to establish a project budget. Most of us choose to use critical path scheduling software, which has been designed to support many of these steps.

The software, itself, has several components. It provides a mechanism for inputting and viewing data. It provides a data storage and management capability. And it provides several algorithms for calculating schedules, resource loads, costs, and variances. While several vendors appear to concentrate primarily on the first two sets of capabilities, it is the latter set of functions that will determine how supportive the software is for generating accurate and useful plans.

This writer has been an outspoken supporter of critical path software for several decades. However, that support comes with a caveat, regarding both the software and the way that it is used. First, the software must allow the user to create an accurate and discrete model of the work and the resources involved in the project. Secondly, the user must be willing to invest the time and effort to affect a usable solution.

The software must allow the user to define just how the work is to be executed, and not force the user to create some artificial plan, just because the tool is too limited to allow finer definition. Scitor's PS7 is an example of a tool that allows finite definition of schedule and resource assignment conditions. For example, the Distribution Spreadsheet Mode allows the user to define exactly how resources are applied to tasks. Additional features support discontinuous application of resources (determined during the resource leveling execution), and assignment to multiple tasks.

*Courtesy of:
Scitor Corporation
256 Gibraltar Drive • Sunnyvale, CA • 94089
800/533-9876*

If this is the way that people work on tasks, then the tool must allow the user to define such conditions, and the user must be willing to take the time to do such. We cannot throw raw data into the machine and allow the computer to come up with an optimal plan. Such a plan can only be achieved through the interaction between a capable software program and an enlightened team of project planners. It is an iterative process that cannot be handed off to a dumb machine.

The Progressing Stage

As we conduct the project, all of the above applies, as we have to be able to replan and adjust to react to execution situations. To this, we must add the ability to input what has taken place and to analyze the impact of performance to date. Here, we can fully appreciate the advances made in database management and multi-source, remote access. But, none of this would be of value if we did not have the tool capabilities discussed above, as well as the dedicated involvement of the project team.

Anyone who says that this function (project statusing) can be automated, to the extent of eliminating or minimizing the involvement of the project team, neither recognizes nor respects the importance of such man-machine interaction in creating and maintaining usable plans.

Concepts and Approaches that Should Be Avoided (or Approached With Caution)

After four decades of involvement with traditional critical path scheduling software, I still strongly support this approach for most applications. There are other workflow-based concepts, both old and new, such as Line Of Balance, and Critical Chain, that have their place. These are not discussed here, due to space limitations. Below are some practices and alternate tool approaches that suggest that the process can be highly automated, and executed without being managed by a project team. Caution is advised. The application of such practices as described below might appear to be alternative project management techniques. But these are merely an illusion of such, failing to support generally accepted project management practices.

AutoActuals

It started over a decade ago, when several of our project management products wanted to give users an easy way of entering actual costs for tasks and resources. They offered an "auto actuals" option, wherein the system calculated the "actual cost" by multiplying the percent complete by the budget. Of course, this meant that the actual costs would always match the planned cost. The Cost Variance would always be zero. Doesn't this defeat the purpose of project cost management? Of course it does. And many project managers rejected systems that offered this "feature".

Automatic Resource Leveling

Automatic resource leveling comes under the caution heading, rather than avoidance. It is a capability that is very important to good scheduling and an expected component of a critical path software package. We must look to our software tools to provide support for resource scheduling, as it is too complicated to do by hand (for the typical project). However, we must draw a line between allowing the computer to create an undirected, unmanaged resource schedule, as opposed to one that is based on directed conditions and management interaction.

It is not that difficult to obtain usable results from resource leveling. But it does require reasonable assistance from a well-designed software package, together with intelligent interaction by the user. For instance:

1. People should not be reluctant to model all of the conditions that would be required to support intelligent, automated resource allocation.
2. The project management software must allow the creation of a discrete assignment model.
3. The resource allocation algorithms must be sophisticated enough to provide acceptable results.
4. The user must interact with the results to fine tune the solution.

Personal Information Managers (PIMs)

Recognizing that traditional project management software is not for everyone, the industry has given rise to alternate approaches toward task and resource planning and management, that are not based on critical path scheduling and serial-mode resource leveling. One of these is the Personal Information Manager (PIM) type of software. Some of these are simple calendar-oriented notebooks -- sort of an electronic Day-Timer. The more sophisticated versions attempt to provide a project orientation to the data in the system. The most recent, and powerful entry into this classification was Team Manager 97, from Microsoft. Yet, it is interesting to note that, despite the excellent design of this product (Team Manager), it has failed to gain any strong acceptance in the marketplace. The project management community has almost totally ignored it (opting instead for traditional PM software) and the rest of the potential market for PIM type software continues to be soft. I will submit the following reasons for this result:

1. For the most part, PIM's are tools that people use for their own information base, rather than as collaborative tools. I guess that's why they call them "personal" information managers. To be used in a project environment, such tools must be standardized, and their use controlled under the direction of an appointed leader. It is not enough to pass information around using such tools. It must be managed.
2. Task information cannot just be changed at will. Changes must be analyzed and either accepted or rejected by the person in charge.

*Courtesy of:
Scitor Corporation
256 Gibraltar Drive • Sunnyvale, CA • 94089
800/533-9876*

3. Individuals cannot always decide what tasks they will work on and when they will work on them. Even with the move to self-directed, multi-discipline teams, we still need to look to the line managers to be involved in staff assignments.
4. In the last 15 years, I have seen several PIM type products brought to the marketplace, that were supposedly optimized for project team applications. None have been successful.

Resource Requestor/Allocator Software

In addition to the PIM nature of Team Manager, a major feature is a designed capability that is supposed to assist in the assignment of resources to tasks. This is a category that I call Resource Request & Allocation Software. The concept is that someone needing work to be done (for instance; a project manager) will populate the database with a list of tasks. These will be communicated to resource owners (presumably the line managers) who will assign resources to the tasks. Here too, we have an interesting concept, that has been tried several times before, and has failed to catch fire. It would be interesting to evaluate the failure of previous attempts to address this resource assignment and modeling need. For instance, why did the Artemis Team product (late '80s) never get off the ground? Why has the latest Artemis effort in this area, Artemis ResourceView, failed to be accepted in the US (and removed from the market)? Why has there been less than a stampede toward Team Manager? Why did Sagacity (Assignment Modeling Method and Software), from Erudite (1990) fall from the face of the Earth? Why has adRem's Project Toolbox, based on an advanced resource allocation method, lacked real success?

I think that the reasons are similar to the PIM situation. In using such tools, we lack a strong project-centric focus on the work, and fail to set up standardized practices under the direction of responsible project and functional leaders. There is no one to evaluate the schedules and assignments and to address issues and conflicts.

Our experience with such tools and concepts tends to reinforce the evidence that traditional critical path techniques, supported by well-designed software and organized project teams is the best way to go.

A Utopian System

More recently, I came across a patent for an "**Automated, Electronic, Network Based, Project Management Server System, for Managing Multiple Work-Groups**". I love that title. It contains all of the popular buzzwords that should get the attention of today's senior managers. The PM equivalent of a low-fat, high-energy, skin-smoothing, muscle-toning, cholesterol-lowering, anti-carcinogen, virility-stimulating soft drink. Heck, you just can't lose with this!

*Courtesy of:
Scitor Corporation
256 Gibraltar Drive • Sunnyvale, CA • 94089
800/533-9876*

Such a computer system (according to selected wording of the patent application) would have the following attributes:

- a. central database
- b. connected to electronic network
- c. using two-way electronic messaging system
- d. storing and accessing data from a multi-project database
- e. automatic in nature
- f. with built-in triggers
- g. based on nature and status of said data
- h. **without need for manual project management coordination**
- i. involves all steps of the PM cycle
- j. identifies owner of received message
- k. identifies nature of received message
- l. setup database - save messages (according to nature of said received message)
- m. receiving project plans and compiling project plans and saving project plans into database
- n. checking plans for resource requests against resource availability and reallocating resources if necessary, based on inter-project priorities
- o. recalculating project plans and sending back said plans based on resource allocations
- p. sending project status reports and reminders to organization work group team members based on status of triggers
- q. receiving project updates and status changes and updating said database
- r. repeating periodically

Wow! Not a bad set of attributes. Wouldn't we all want something like this? Perhaps ... until we notice the magic phrase "***without need for manual project management coordination***". This system would have us eliminate project managers or even project coordinators. Heck, in the attempt to have flatter organizations, maybe we can do away with resource managers, as well. Project plans would somehow find their way into this database, automatically. Then, the system would communicate work commitments and work status, and would be continually updated via electronic messaging. The "system" would automatically resolve all conflicts ... without requiring human intervention.

And here I thought that I would never find a utopian project management environment in my lifetime. Beam me up, Scotty!

Courtesy of:
Scitor Corporation
256 Gibraltar Drive • Sunnyvale, CA • 94089
800/533-9876

Hey, I don't want to knock the objectives of such automation. It's just that we can't fantasize about removing the human element. Computer systems cannot manage projects or work-groups. They can only offer aid to the people charged with the responsibility to deliver project and operational results. We need sophisticated tools that will make it easier to plan and track work, to evaluate the impact of new work, to evaluate the progress of existing work, to help with the assignment of scarce and shared resources, etc.

New products will continually appear that will appeal to those who wish that this could be accomplished without dedicated managers and organizations. These products will be rejected because they don't work. These products will be rejected because they support a fallacious dream of work being managed and conflicts being resolved by a computer. The result will always be a nightmare, instead. I have an entire room full of such products, most of which are no longer being sold.

Harvey A. Levine, with 38 years of service to the project management industry, is founder of **The Project Knowledge Group**, a consulting firm specializing in PM training, PM software selection, evaluation & implementation, and PM using microcomputers.

He has implemented or enhanced the project management capabilities of numerous firms, often combined with the selection or implementation of computerized project management tools. Mr. Levine is considered the leading consultant to the project management software industry and is recognized as the leading expert in tools for project management.

He has been an Adjunct Professor of Project Management at Rensselaer Polytechnic Institute and Boston University. And has conducted numerous project management public seminars for ASCE, AMA, IBM, and PMI.

Mr. Levine is the author of the book "Project Management using Microcomputers", and has been published extensively in other books, periodicals and videos.

Mr. Levine is a past president of the Project Management Institute and the recipient of *PMI's 1989 Distinguished Contribution to Project Management* award. Recently, he was recently elected as a *Fellow of PMI*.

Mr. Levine has offices in Saratoga Springs, NY and San Diego, CA and can be contacted via e-mail at: LevineHarv@cs.com